Docket No.: 2012(220768)
(PATENT)

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent Application of:
Pawel A. Chadzynski

Application No.: 10/757,975

Confirmation No.: 4258

Filed: January 15, 2004

Art Unit: 2193

For: SYNCHRONOUS AND ASYNCHRONOUS COLLABORATION BETWEEN HETEROGENEOUS APPLICATIONS

Examiner: Vu, Tuan A.

SUPPLEMENTAL CLAIM AMENDMENTS FOR ENABLING EXAMINER'S AMENDMENT

MS Amendment
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir:

INTRODUCTORY COMMENTS

In response to the telephonic interviews conducted with Examiner Tuan on September 23, 2009 and September 28, 2009, Applicant respectfully proposes the following claim amendments for entry as an Examiner's Amendment:

A **Listing of the Claims** begins on page 2 of this paper.

Remarks/Arguments begin on page 13 of this paper.

LISTING OF THE CLAIMS

1. (Currently amended) A computerized method for synchronization of a plurality of heterogeneous user applications running on respective clients collaborating over a network via a session manager to manipulate a design manipulated by a plurality of users representing electrical or mechanical assemblies ~~using a plurality of heterogeneous user applications running on respective clients connected to the network~~, said method comprising the steps of:

connecting a session client process to a session manager over the network to participate in a collaborative session;

sharing session control messages with other session client processes connected to said session manager;

loading design data representing said design into a local application running on said client;

creating at least one application state file representing at least one application state of said local application based on at least one manipulation of said design using said local application;

communicating said at least one application state file from said session client process to said other session client processes via said session manager;

receiving a first ~~at least one~~ application state file created by another user on another local application[[s]] and communicated from one of said other session clients via said session manager;

presenting the first ~~at least one~~ application state file created by another user on another local application[[s]] to a user of the session client process to allow the user to refuse or delay loading of the first application state file; and

if the user applies to delay loading of the first application state file:

buffering the first ~~at least one~~ application state file at the application layer to allow the user to incorporate the buffered first application state file into the design data at a later user-determined time; and

~~delay the instantiation of~~ loading the first ~~at least one~~ application state file created by the other users on the other local applications ~~until~~ at a time determined by the user of the session client process;~~ and~~

~~loading the at least one application state file created by other local applications,~~

wherein the buffering of incoming application state files enables the user of the client process to selectively apply other users' changes to the design, thereby allowing the user of the session client process to manipulate a first aspect of the design before loading changes made to a second aspect of the design by the ~~an~~other users within the other users' respective local applications.

2. (Original) The method of claim 1 wherein said at least one application state is encoded using normalized XML structures to create said at least one application state file, and wherein said at least one application state file is communicated as an XML message.

3. (Original) The method of claim 2 wherein said XML structures are based on domain specific conventions defined in the context of the type of design data.

4. (Original) The method of claim 1 further comprising saving said session controls and said at least one application state file in a journal file.

5. (Original) The method of claim 1 further comprising the step of scheduling said collaborative session.

6. (Original) The method of claim 1 further comprising the step of conducting a text-based conversation with said other session clients.

7. (Original) The method of claim 1 further comprising the steps of logging in to said collaborative session and logging out of said collaborative session.

8. (Canceled)

9. (Original) The method of claim 1 further comprising the step of displaying design manipulations corresponding to said application state file created and communicated by said other application files.

10. (Original) The method of claim 1 wherein said design is manipulated without having to transmit design images between said heterogeneous applications.

11. (Currently amended) A computerized method for providing asynchronous training to a user to manipulate a design representing electrical or mechanical assemblies using a plurality of heterogeneous applications, said method comprising the steps of:

loading design data into a local application on a computer;

manipulating a design based on said design data using said local application;

creating at least one application state file representing an application state of said local application based on at least one manipulation of said design using said local application;

saving each said at least one application state file in a journal file; and

transmitting said journal file to another computer such that said at least one application state file can be loaded on said another computer and said at least one application state file ~~manipulation~~ can be presented to the user to allow the user to refuse or delay review~~ed~~ of said at least one application state file using another heterogeneous application running on said another computer, wherein said journal file provides interactive instructions when played back on said another computer; and

if the user chooses to delay loading of said at least one application state file:

buffering said at least one application state file at the application layer to allow the user to incorporate the buffered first application state file into the design data at a later user-determined time; and loading said at least one application state file at a time determined by the user;

wherein the buffering of application state files enables the user to selectively apply other changes to the design, thereby allowing the user to manipulate a first aspect of the design before loading changes made to a second aspect of the design and providing asynchronous training to the user.

12. (Original) The computerized method of claim 11 wherein said method is an asynchronous method of collaboration.

13.-16. (Canceled)

17. (Currently amended) A computer program product comprising a storage medium having computer-readable code stored thereon synchronization of a plurality of heterogeneous user applications running on respective clients for collaborating over a network via a session manager to manipulate a design manipulated by a plurality of users representing electrical or mechanical assemblies~~ using a plurality of heterogeneous applications running on clients connected to the network~~, said system comprising:

at least one application state file for representing application states created by a local application based on at least one manipulation of a design made using said local application; and

a session client module for synchronizing interaction between said local application and a collaborative session and for communicating said application state file over the network to other clients and receiving at least one other application state file created by other users from the other clients, presenting the at least one other application state file to a user of the computer program product to allow the user to refuse or delay loading of the at least one other application state file, and if the user chooses to delay loading of the at least one other application state file:

buffering the at least one application state file at the application layer to allow the user to delay the user to incorporate the at least one application state file into the design data at a later user-determined time;

loading the at least one application state file created by the other users on the other clients at a time determined by the user of the computer program product;

wherein the buffering of incoming ~~instantiation of the at least one other~~ application state files enables the user of the computer program product to selectively apply other users' changes to the design ~~until a time determined by the user of the session client module, and loading the at least one other application state file~~, thereby allowing the user of the computer program product to manipulate a first aspect of the design before loading changes made to a second aspect of the design by the ~~an~~other users within the other user's respective local computer program products.

18. (Previously presented) The computer program product of claim 17 wherein said session client module comprises:

a user interface for interfacing with a user;

a first application programming interface for interfacing with said local application; and

a second application programming interface for interfacing with a session manager over the network.

19. (Previously presented) The computer program product of claim 17 further comprising a session journal for recording session controls and said application states during said collaboration session.

20. (Previously presented) The computer program product of claim 17 wherein said application states are encoded using normalized XML structures to create said at least one application state file, and wherein said at least one application state file is communicated as an XML message.

21. (Previously presented) The computer program product of claim 17 wherein said session client module communicates said application state file without sending images of said design.

22. (Currently amended) A computer program product comprising a storage medium having computer-readable code stored thereon for providing asynchronous training to a user to manipulate a design representing electrical or mechanical assemblies using a plurality of heterogeneous applications, said system comprising:

means for loading design data into a local application on a computer;

means for manipulating a design based on said design data using said local application;

creating at least one application state file representing an application state of said local application based on at least one manipulation of said design using said local application;

means for saving each said at least one application state file in a journal file; and

means for transmitting said journal file to another computer such that said at least one application state file can be loaded on said another computer and said at least one application state file ~~manipulation~~ can be presented to the user to allow the user to refuse or delay review~~ed~~ of said at least one application state file using another heterogeneous application running on said another computer, wherein said journal file provides interactive instructions when played back on said another computer, and if the user chooses to delay loading said at least one application state file:

buffering said at least one application state file at the application layer to allow the user to incorporate the buffered first application state file into the design data at a later user-determined time; and

loading said at least one application state file at a time determined by the user;

wherein the buffering of application state files enables the user to selectively apply other changes to the design, thereby allowing the user to manipulate a first aspect of the design before loading changes made to a second aspect of the design and providing asynchronous training to the user.

23. (Currently amended) A computer program product comprising a storage medium having computer-readable code stored thereon for synchronization of a plurality of heterogeneous user applications running on respective clients collaborating over a network via a session manager to manipulate a design manipulated by a plurality of users representing electrical or mechanical assemblies~~ using a plurality of heterogeneous applications on client computers connected to the network~~, said computer-readable code comprising:

code for interfacing with a user;

code for interfacing with a local application used to manipulate a design;

code for interfacing with a session manager over a network;

code for creating a session client process to communicate session controls and at least one application state file to said session manager over said network, wherein said application state file is created by said local application based on at least one manipulation of said design; and

code for receiving at least one other application state file created by another user on other local applications from said session manager over said network, presenting the at least one other application state file created by other users on another local application to the user to allow the user to refuse or delay loading of the first application state file, and if the user chooses to delay loading of the at least one other application state file:

buffering the at least one application state file at the application layer to allow the user to incorporate the buffered first application state file into the design data at a later user-determined time; and

~~delay the instantiation of~~ loading the at least one other application state file created by the other users on the other local applications at ~~until~~ a time determined by the user of the session client process, wherein the buffering of incoming application state files enables the user of the client process to selectively apply other users' changes to the design ~~and loading the at least one other application state file~~, thereby allowing the user of the session client process to manipulate a first aspect of the design before loading changes made to a second aspect of the design by the ~~an~~other users within the other users' respective local applications.

24. (Currently amended) A computer program product comprising a storage medium having computer-readable code stored thereon for synchronization of a plurality of heterogeneous user applications running on respective clients collaborating over a network via session manager to manipulate a design manipulated by a plurality of users representing electrical or mechanical assemblies~~ using a plurality of heterogeneous applications on client computers connected to the network~~, said computer-readable code comprising:

code for interfacing with a user;

code for interfacing with a local application;

code for instructing said local application to create at least one application state file representing at least one application state based on a manipulation made to a design using said application state file;

code for interfacing with a session manager over a network;

code for sending said at least one application state file and session controls to said session manager; and

code for notifying said local application that at least one other application state file created by other users on other local applications has been received, presenting the at least one other application state file created by other users on other local application to the user to allow the user to refuse or delay loading of the first application state file, and if the user chooses to delay loading of the at least one other application state file:

buffering the at least one application state file at the application layer to allow the user to incorporate the buffered first application state file into the design data at a later user-determined time; and

~~delay the instantiation of~~ loading the at least one other application state files created by the other users on the other local applications at ~~until~~ a time determined by the user; [[,]]

wherein the buffering of incoming application state files enables the user to selectively apply other users' changes to the design ~~and loading the at least one other application state file~~, thereby allowing the user to manipulate a first aspect of the design before loading changes made to a second aspect of the design by the ~~an~~other users within the other user's respective local applications.

25. (Original) The computer program product of claim 24 further comprising code for controlling instantiation of said received application state file into said local application.

26. (Currently amended) A computerized method for synchronization of a plurality of heterogeneous user applications running on respective clients collaborating over a network via a session manager to manipulate a design manipulated by a plurality of users representing electrical or mechanical assemblies ~~using a plurality of heterogeneous user applications running on respective clients connected to the network~~, said method comprising the steps of:

connecting a session client process to a session manager over the network to participate in a collaborative session;

sharing session control messages with other session client processes connected to said session manager;

loading design data representing said design into a local application running on said client;

creating at least one local application state event representing at least one application state of said local application based on at least one manipulation of said design using said local application; and

communicating said at least one local application state event from said session client process to said other session client processes via said session manager;

receiving at least one other application state file created by other users on other local applications and communicated from said other session clients via said session manager; [[,]]

presenting the at least one other application state file created by another user on another local application to a user of the session client process to allow the user to refuse or delay loading of the at least one other application state file; and [[,]]

if the user chooses to delay loading of the at least one other application state file:

buffering the at least one application state file at the application layer to allow the user to incorporate the buffered at least one other application state file into the design data at a later user-determined time; and

~~delay the instantiation of~~ loading the at least one other application state file created by the other users on the other local applications at ~~until~~ a time determined by the user of the session client process;

wherein the buffering of incoming ~~, and loading the at least one other~~ application state file enables the user to selectively apply other users' changes to the design, thereby allowing the user to manipulate a first aspect of the design before loading changes made to a second aspect of the design by the ~~an~~other users within the other users' respective local applications.

27. (Previously presented) The method of claim 26 wherein the step of communicating further comprises:

communicating said at least one local application state event as at least one data packet for representing said at least one application state event.

28. (Previously presented) The method of claim 26 wherein the at least one local application state event is at least one of a plurality of normalized application state events recognized by each of the heterogeneous user applications.

29. (Previously presented) The method of claim 26 further comprising the step of:

saving said session control messages and said at least one local application state event in a journal file.

30.-31. (Canceled)

REMARKS

Claims 1-7, 9-12, and 17-31 are currently pending. Claims 1, 11, 17, 22-24, and 26 are currently amended. Claims 30 and 31 are canceled. Accordingly, claims 1-7, 9-12, and 17-29 will be pending after entry of this amendment.

Support for the amendments is found throughout the specification and claims as originally filed. For example, support for the amendments to claims 1, 11, 17, 22-24, and 26 is found in ¶ [0038] of the specification as originally filed. No new matter is added.

Telephonic Interviews

Applicants thank Examiner Tuan for speaking with Applicant's representatives, Ralph Loren and Brian Landry, on September 23, 2009 and September 28, 2009. During the telephonic interviews, Applicant's representatives discussed the proposed amendment herein to claim 1. At the conclusion of the telephonic interview, Examiner Tuan generally agreed to the proposed amendment and requested that Applicant propagate the amendments throughout the application.

Conclusion

In view of the above amendments and remarks, Applicant believes the pending application is in condition for allowance.

Applicant believes that no fees or extensions are required other than fee for an Information Disclosure Statement under 37 C.F.R. § 1.17(p). However, if for any reason the authorized fee is inadequate, the Office is conditionally authorized and requested to charge Deposit Account No. **04-1105** under order number 2012(220768). Also, the Office should consider this a conditional petition for the proper extension period needed to have this response entered and considered, if any.

Dated: September 28, 2009

Respectfully submitted,

Electronic signature: /Brian R. Landry/
Ralph A. Loren
Registration No.: 62,074
Ralph A. Loren
Registration No.: 29,325
EDWARDS ANGELL PALMER & DODGE LLP
P.O. Box 55874
Boston, Massachusetts 02205
(617) 239-0233
Attorneys For Applicant

Customer No.: 21874